

FACTA UNIVERSITATIS

Series: **Automatic Control and Robotics** Vol. 18, N° 1, 2019, pp. 31 - 40

<https://doi.org/10.22190/FUACR1901031V>

## SCIENTIFIC REFERENCES IMPORT FROM UNSTRUCTURED DATA

UDC ((004.738.2+004.41):655.535.2)

**Nataša Veljković, Darko Puflović, Leonid Stoimenov**

University of Niš, Faculty of Electronic Engineering, Computer Science Department, Niš,  
Republic of Serbia

**Abstract.** *Along teaching, scientific research is the second most important task for every university and faculty member. Research results are commonly published as journal papers, book chapters or as conference proceedings with the purpose to communicate the findings in academic circles, serve as promotion, or be used for tenure or reapplication. Maintaining the list of scientific references can be a time consuming and tedious activity. To help along the process we have created a Web application named References that makes the process of editing and formatting the references partially automated and less time consuming. In this paper we present the architecture of the application, its functionalities and a newly added feature that enables importing references both from structured and unstructured data.*

**Key words:** *bibliography, references management software, unstructured data*

### 1. INTRODUCTION

Fifty years ago, publishing research results in journals and monographs and attending conferences to present new ideas was the only way to promote and share a research. Nowadays, the Internet has become a preferred way of communication and publishing, even in Academia. We can publish papers online, find sources for the research, browse among thousands of papers or track citations and bibliographies of other researchers and scholars.

By definition, bibliography is a list of references on knowledge used during the research or in the preparation of a scientific paper [1]. It has become a common thing for a researcher to maintain a personal bibliography which can be used as the means of self-promotion or for reapplication or tenure on Universities. This list can become hard to maintain and format if kept as the plain text, but there are software products that can ease

---

Received February 01, 2019

**Corresponding author:** Nataša Veljković

Faculty of Electronic Engineering, Computer Science Department, Niš, Republic of Serbia

E-mail: [natasa.veljkovic@elfak.ni.ac.rs](mailto:natasa.veljkovic@elfak.ni.ac.rs)

the process of references management. Researchers and teaching staff at the Faculty of Electronic Engineering of the University of Niš, have been using collaborative reference management software - *Reference*, since 2008. It started as a student project, but it grew into a complex software that provides collaborative reference management, easy generation of references' list, search of publications, visual analysis of references, export of references into common formats, export using predefined and custom styles, and import of references [2, 3].

On the Web we can find scientific references data both in structured and unstructured formats. Structured formats allow us to import and consume references efficiently, to collect information using predefined terms, about a certain author, category, year, etc. or connect the information about authors, papers and journals in the organized manner. On the other hand, unstructured reference data are not organized in the predefined manner and cannot be recognized with high precision using most of the tools available. There are tools that can be used to identify parts of the unstructured and structured references and provide conversion to the structured formats, but most of them can be used only to parse a certain style or convert it into some other widely used one. Existing solutions can be divided into categories by the technique they use to determine the results or format of data accepted as input. When it comes to the format of the input, most solutions accept pdf files, but there are solutions that can be used as a Web service or Desktop application and can parse reference from plain text. Techniques used can divide available tools into categories that use the rules based approach [4-7] and machine learning approach [8-14]. Machine learning tools can be used to perform a better analysis of the reference, with a potential to learn from the parts of the reference that initially cannot be recognized, that makes it better for unstructured data parsing. To make the *Reference* software more user-friendly, especially for users with many published papers, we have developed a Web service that can recognize and import references from structured and unstructured data and connect them to the existing authors in the system. The application accepts an URL to the structured references data, such as Open Researcher and Contributor ID (ORCID) [15] or an URL to the unstructured html page or pdf document on the web that has information on published papers but no metadata describing different reference elements, as a result we provide recognition and import of found references into our software.

This paper is organized as follows. It begins with a description of related work on a different reference management software. Section 3 describes Reference application architecture, module and functionalities. The process for identifying bibliography references in a free text is described in Section 4. The methodology used in Reference application to import data from an unstructured text is described in Section 5. Conclusion and ideas for the future work are presented in Section 6.

## 2. REFERENCES MANAGEMENT SOFTWARE

Software for scientific references management has been present for many years now, and in general we can divide it on the Web based and Desktop based solutions. In fact, the analysis of free Web based tools that create citations and format bibliographies, given in [16], has proven that Web based solutions can be used as effectively and accurately as their Desktop matches. To compare with *Reference* application we will analyse the features of the existing most popular non-commercial Web based solutions.

Rebase [16, 17] enables management of bibliographic references, citations and scientific literature. It is designed to create collections of bibliographic references imported from different academic databases and to extract them in quotation form for text bibliographies [18]. It offers multitude of search options, saving search query for future usage, email announcements for newly added records, export of records in various formats.

Aigaion [19] is designed to enable management of annotated bibliographies. What distinguishes Aigaion from other reference management tools is the possibility to categorize publications in a topic tree or by using keywords [20]. Users can subscribe or unsubscribe from topics that are relevant to them. Each author and topic display offers a keyword cloud showing the keywords that are assigned to the publications of the author or topic. Besides this, Aigaion offers standard functionalities such as user management, search and import/export of publications from/to standard formats.

RefWorks [21] is a commercial Web-based software that can be used for free during a 30 days of trial period, after that the user has to choose the type of subscription. Refworks was built as a tool for academic libraries [22] but has proven to be helpful to researchers as well [23]. It enables users to gather, manage, store and share all types of information, as well as generate citations and bibliographies.

Wikindex [24] enables management of bibliographies, quotations and articles. It can support either single use or multi-user collaborative usage. It offers users management, articles and notes creation and management, searchable references, notes and citations, import and export to other bibliographic formats including BibTeX, Reference Manager, ProCite and EndNote.

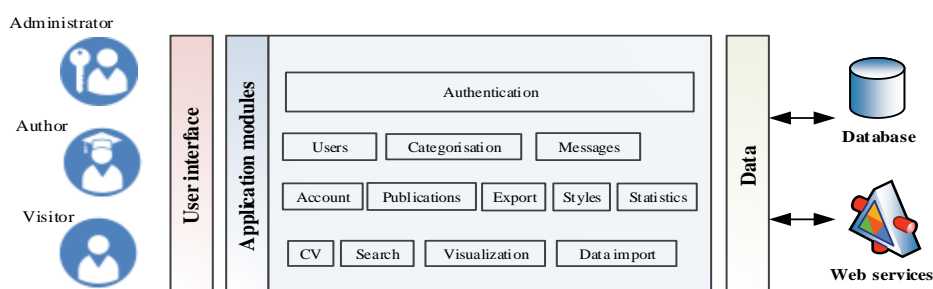
Memento bibliographic software [25] is built for students, researchers and scholars. It features import and export of articles in multiple formats, RSS feeds, creation of reports, sharing reports with other researchers, creation of watch lists for articles, authors or keywords.

Another interesting product is the Open Researcher and Contributor ID (ORCID) [15]. The ORCID provides a persistent digital identifier for authors aimed to solve the ambiguity problem in research and scholarly publications. ORCID platform uses Crossref metadata search [26] and Scopus [27] in order to import and link papers to the author's account. The author can also add papers that are not found using these two search engines. This is a powerful tool that can quickly collect the author's work in one place.

Given the overview of these Web based software products, we can note the obvious common characteristics: users' management, references management, import/export of references, search and notifications. Some interesting features like articles or annotations management (Rebase, Wikindex), topic and categories based categorization (Aigaion) can add more value to the software and make it more appealing. It is interesting to note that all of the aforementioned products support the import or export of references into common formats, such as BibTeX or RIS, however, they do not have the possibility to import references from unstructured plain text or from the HTML page. Formatting of references is supported using only predefined styles (MLA, Chicago, APA, etc.) while user generated styles are not supported. The categorization of references is predefined and not manageable via the application. In the single-user mode, one can adjust the display to see papers by keywords or topics, but there is no possibility to administer the categories of references that are common for all users. Also, one important feature that is missing in all the software is the visual analysis of scientific references.

### 3. REFERENCES WEB APPLICATION

Reference application architecture (see Fig.1.) is a three-layered architecture consisted of: user interface, application logic layer and data layer [2]. There are three types of users graduated by the level of control over the application: visitor, author and administrator. Visitors are users that cannot modify the existing or enter new scientific references, they have only read privileges and the possibility to search references. Authors and administrators have both read and write privileges. Administrator is in charge of managing users and categories of scientific references. Author can manage its publications and use other functionalities of the application associated with the search, visualization, custom page display and the import/export of references. Visitor can only search publications, browse co-relational graphs and publicly available CV's of authors.



**Fig. 1** Web application architecture

The application logic is realized using separate modules that can manage user activity and connect different application layers. Depending on the user type, these modules are passive or active. The list of modules and their roles is presented in detail in Table 1.

**Table 1** Modules in Reference application

Name	Functionality	Visitor	Author	Administrator
Authentication	manages user login and page access rights	✓	✓	✓
Users	manages user accounts			✓
Categorisation	manages scientific categories			✓
Messages	sends notifications to the authors			✓
Account	manages author related data		✓	
Publications	manages scientific publications		✓	
Export	exports publications in different formats		✓	
Styles	creation of custom bibliographic styles		✓	
Statistics	generates statistics for the author		✓	
CV	generates custom html page for the author		✓	
Search	search references using different criteria	✓	✓	✓
Visualization	generates co-relation graphs	✓	✓	✓
Data import	imports data from external sources		✓	

The authentication module is active all the time. It takes care of user-login and Web application access rights. Module Users manages user accounts by supporting the administrator with functions to add, edit or delete users. The Categorization module enables adding, editing or deleting the categories of scientific references. The Messages module enables sending email notifications to registered users. The Account module takes care of

the change of personal data and login information for authors. The Publications module allows authors to add, update or delete their publications. The Export module enables export of publications into common formats: BibTex, RIS, Microsoft Office 2007 XML, TXT or PDF or their output to the connected printer. The Styles module controls generating author's custom styles for reference formatting and enables the usage of application defined styles. The Statistics module generates basic statistics about entered publications for each author. The CV module enables creation of publications list formatted using previously created or application defined styles. The Search module enables references search by author, title of publication, category and the year of publication. The Visualization module depicts relations between different entities (authors, publications, keywords, etc.) to enable efficient interpretation and analysis of scientific references' data.

The Data layer enables access to database and external services. The Local Web application's database stores different data on registered users and publications. The application currently uses three internal services:

- Web service for export of data to interchangeable formats (BibTex and RIS)
- Web service for the visual analysis of references
- Web service for importing references from external sources

Each publication entered in the system can be associated to one of the following categories [28]: publication in journals (M50, M20), monographs (M10 and M40), conference proceedings (M30 and M60), technical solutions (M80) and thesis (M70). Each category is further divided into sub-categories, for example: paper presented on the international conference printed entirely (M33), paper presented on the international conference printed as abstract (M34), paper presented on the national conference printed entirely (M63), paper presented on the national conference printed as abstract (M64), etc. Categories can be added or modified by a user with Administrator role. During customization, Administrator can even define which fields will be shown in every subcategory.

The following predefined styles exist in the *Reference* application: Default style (used for internal references formatting at the Faculty of Electronic Engineering), MLA, APA and Chicago. There is also a possibility to create a custom style format (see Fig.2.), which

**Kreiranje stila**

Dodavanje novog elementa: [ , [ , ] , { , } , \* , & ]

Odvajanje svakog od elemenata sa: [ , [ , ] , { , } , \* , & ]

Formatiranje elementa: **Bold** *Italic* Underline

Uklanjanje formatiranja sa elementa: **Bold X** *Italic X* Underline X

Učitavanje stila: Podrazumevani APA Chicago

Resetovanje šablona: Resetuj šablon

Ime autora	SS	Prezime autora	Inicijal imena autora	Inicijal prezimena autora	Autor2. N	AutorN	Naslov izdavač	Volumen	Strana	Broj	Časopis
Edicija	Konferencija	ISSN/ISBN	Godina	Period	Mesto	Editori	Internet adresa	DOI			

Naziv stila  .xml

**Kreiraj stil**

**Fig. 2** Creating custom style for references

is extremely useful when using data import from an unstructured text. It is available only for Authors, but created style template by one Author can be shared and used by other Authors as well. Author creates a style template by choosing the order of blocks, for example, the author's name can be set as the first block, and the surname as the second, the year of the publication can be added as the third block and so on. If some block is not needed, it can be removed from the list. Blocks are moved easily on click and drop, and they are removed by clicking on the x in the top right corner. When blocks are aligned in the desired order, the author chooses the separators of the blocks which can be: comma, period, space, open bracket, closed bracket, quote, double quote, and dot. Separators can be added one by one or they can be applied to all blocks at once. Instead of creating a template from scratch, a user can pre-load one of the existing templates, modify it and save it as a new template. A Template is saved as an .xml file which is interpreted by the application when the user needs to apply selected style to references list.

The usage of the application in the Author's role is currently restricted only to the members of the Faculty of Electronic Engineering. Registration is not available from the main Web page (<http://cs.elfak.ni.ac.rs/reference>), but an administrator can create an Author on request. In this way, the application cannot be used outside of specified domain. However, it is not limited in this way and it can be used for any other University or Faculty member.

#### 4. IMPORTING REFERENCES FROM UNSTRUCTURED DATA

References can be found in the variety of styles, some of which are custom made, and can be difficult to recognize and parse, but every reference, no matter the style, has to contain the elements that can identify and describe it, like author names, title, year, volume, number, pagination, DOI and etc. An example of different styles and elements of bibliographic references can be seen in Table 2. Different citation systems and styles are being used in different disciplines because they work better with the kind of texts that are most commonly used in that discipline. For example, APA is used in education, psychology and sciences, MLA in humanities, while Chicago is generally used in business, history and art [29]. Those styles are used to format bibliography which can be considered as structured text because it follows certain rules. On the other hand, unstructured text is not standardized or organized in a predefined manner, it is defined freely by the person who published it.

Unstructured references can be processed in many different ways, depending on the information available about the format used. If that information is available, it is possible to use regular expressions to extract valuable information needed, but in some cases that format is not known in advance. Information collected during the parsing of structured data can be used to understand the unstructured one better [30]. Every piece of information that can be found in this process can be valuable to further improve the understanding of text, like the style of text used, font weight and style, special characters that can be found in specific places, etc. Once used to understand the rules used to create references, those specific details about the structure can improve recognition and make better predictions about unrecognized pieces of text.

**Table 2** Bibliographic reference presented using different styles

Style	Example
MLA	Marino, José B., et al. "N-gram-based machine translation." <i>Computational linguistics</i> 32.4 (2006): 527-549.
APA	Marino, J. B., Banchs, R. E., Crego, J. M., de Gispert, A., Lambert, P., Fonollosa, J. A., & Costa-Jussà, M. R. (2006). N-gram-based machine translation. <i>Computational linguistics</i> , 32(4), 527-549.
Chicago	Marino, José B., Rafael E. Banchs, Josep M. Crego, Adrià de Gispert, Patrik Lambert, José AR Fonollosa, and Marta R. Costa-Jussà. "N-gram-based machine translation." <i>Computational linguistics</i> 32, no. 4 (2006): 527-549.
Harvard	Marino, J.B., Banchs, R.E., Crego, J.M., de Gispert, A., Lambert, P., Fonollosa, J.A. and Costa-Jussà, M.R., 2006. N-gram-based machine translation. <i>Computational linguistics</i> , 32(4), pp.527-549.
Vancouver	Marino JB, Banchs RE, Crego JM, de Gispert A, Lambert P, Fonollosa JA, Costa-Jussà MR. N-gram-based machine translation. <i>Computational linguistics</i> . 2006 Dec;32(4):527-49.
Custom style	J.B. Marino, R.E. Banchs, J.M. Crego, A. Gispert, P. Lambert, J. Fonollosa, M. R. Costa-Jussà, N-gram-based machine translation, <i>Computational linguistics</i> , 32 (4), 527-549, 2006.

Reference parsing is no different than other named entity recognition tasks which are greatly used in natural language processing. When it comes to recognizing references, existing approaches are using either rules or machine learning techniques. Mostly because the unstructured text that contains references is not random as much as the text used for other named entity recognition tasks that makes the process of recognizing reference parts easier. The parts of the text that can pose challenges to the accurate parsing of references are author names, publication title, journal in which the work was published, volume, issue, year of publication, DOI, page range and other less commonly used entities like the date on which the article was last time accessed, URL and other reference metadata. There can be different formats for author names, for example: last name followed by initials (Harvard, Vancouver and APA), the last name followed by the middle name initials and the first name (MLA, Chicago), and only the first author is listed followed by et al. (MLA). Article titles can be under quotation marks (MLA, Chicago) or without them. The publication year can appear after the list of authors (APA, Harvard), after the article title (Vancouver) or before pagination details (MLA, Chicago) and it can be entered within parentheses. Pagination may be preceded by "pp.", "p.", or semicolon. There are also few combinations of volume and number, and there can be minor variations in the use of commas, spaces, semicolons or dots to separate different elements. Also there might be a custom style used on the page for which we do not know the order of elements before processing.

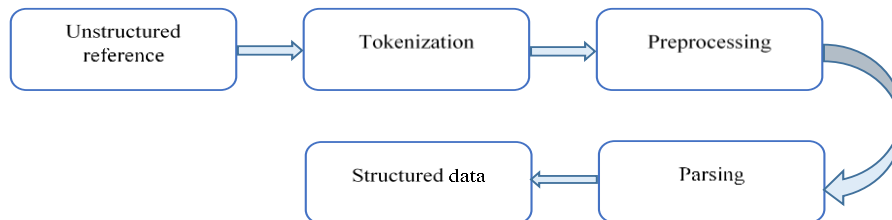
Rule-based methods require the domain expert's knowledge to prepare and maintain the rules over time. They can be very successful for journals that use the same publishing style for references, but require additional tuning for usage with a heterogeneous set of journals [7]. Very often, rules and templates are combined together. Templates summarize patterns in the text surrounding the reference or the reference itself, when the text matches the template entities extraction is performed using the rules [8]. Day et al. [9] focused on a knowledge based approach and created an ontological framework called INFOMAP that can extract reference entities from a natural language text.

Machine learning approaches focus on building the knowledge on reference styles and templates from the training samples. They outperform rule-based parsers on average due to the

fact that it is easy to have good precision on custom defined rules, but it is difficult to manually create large number of rules that will cover different reference styles [10]. Mostly used machine learning classifiers include Hidden Markov Model (HMM), Conditional Random Fields (CRF) and Support Vector Machine (SVM). In the work of Cui and Chen, HMM is used to extract metadata from PDF scientific documents [11], while Seymore et al. use this classifier to extract entities from the research paper headers [12]. Councill et al. implemented an open source package for parsing references using CRF classifier [13]. Okada et al. used SVM to automatically identify reference entities (author, title, journal and year) and then applied HMM to improve the accuracy of references recognition [14].

## 5. THE METHODOLOGY

The process of parsing the reference from the unstructured text in the *Reference* application starts with the tokenizer (see Fig.3.) that will break the text into tokens (word, phrase, symbol) that contain valuable information. These tokens will be used as subsequent input for the pre-processor. Reference consists of information needed in the parsing process, but also a number of punctuations that can be misleading in the next steps if preserved. Author names are separated using commas, or combination of punctuations. It often happens that the last author has & (and) conjunction before the name. When recognizing references, the conjunction is an undesirable element that can only produce errors in the further processing, so it has to be removed. On the other hand, there are elements that are essential, such as *vol.*, *no* or *pp* because they are followed by the value for which they provide meaning. Other elements of reference can be recognized one part at the time. During the process it is important to remove the recognized parts since this helps in making the process faster and more accurate.



**Fig. 3** Process of transforming unstructured text into structured data

Because the process of tokenization does not have information about the structure of the specific reference parts, but only the list of all the separators, it is possible that the separation goes a step further and breaks the pieces of the structure that can be of interest. The solution to this problem is to preserve information about the initial order of the text which can help to reconnect the broken pieces and still recognize them as a whole. Using similar methods to the ones used to parse structured data, now it is possible to recognize each part of the text as any of the possible formats used to represent data in every other reference style used. Excluding the recognized items inside of the text, but also the order in which those items appear in unstructured data, can further help making the recognition process more accurate. The pre-processing of the tokenized unstructured data is a necessary step to ensure that the process of parsing is going to be error prone and that the parts of the text that contain errors



are excluded or fixed. The limited number of formats in which references exist helps making models that can be used to recognize all of the known formats using only simple techniques, like n-gram models or even regular expressions.

However, the parsing and recognition process have to be a combination of those methods and other named entity recognition tasks that lowers the number of unrecognized parts to the minimum. Certain reference parts can benefit from the information that can be stored beforehand and updated every time new information appears. Author names are one of those parts that can be recognized with ease if the information about all authors is available to the parser before detecting names without that information. Not only can a database of author names help find an author more easily, but it can also help make that process more accurate and the final result can be with more metadata, like the complete first and middle name, even if that information is not stored inside the reference itself. Other elements of references that can benefit from the information stored beforehand are numbers and abbreviations. Year of publication, dates, pages and abbreviations that are used to determine what each of those numbers represent are only few examples of what can be accomplished if additional information about reference formats are available to the parser.

Depending on the format, it is possible that in the end of the parsing process there are still parts of the text that are not recognized. Using them to determine what class they belong to can further help improve the process, by adding new categories inside the list of the known ones that can help in the future analysis. That makes the process of recognition more flexible and able to adapt to the new formats which can be useful in the cases where references contain large amount of metadata, but also in the cases where references are formatted in the previously unknown format.

Structured data is formed from the following recognized entities: Author Names, Article Title, Journal (Chapter, Conference, Web source), Volume, Issue, Pagination, Year of the publication, DOI, ISSN (ISBN), Web page link and Other. The reference is presented to the Author who can alter data and the save reference.

## 6. CONCLUSION

Software tools for references management simplify the process of locating and importing citation details and enable efficient and timely management of references. Among number of solutions for references management, Web solutions stand out for the simplicity of usage, and the possibility for a user to manage references without time and place constraints. All described solutions have their good sides and justification for usage, but what singles out our solution is the possibility for collaborative references sharing and maintenance. Reference Web application is a simple solution for a research group to share and manage common bibliographic database on the Web. An up-to-date list of references is available to every researcher at any time. References can be used as is or they can be exported in any common format or bibliographic style. Research can be shared easily via customizable personal page, it can be searched and visualized using different types of graphs. Reference import from the unstructured data is a powerful extension to the existing management system, especially when transferring existing data from the systems that use unstructured data for storage. However, the complexity of the problem can use further research to make the entire process more robust and precise, with even more recognizable features. The database of the existing papers and authors can be used more

efficiently in combination with online services and data, to extract even information that does not necessarily exist inside the unstructured data itself.

## REFERENCES

- [1] B. Gastel, R.A. Day, How to write and publish a scientific paper, ABC-CLIO, 2016.
- [2] N. Veljković, I. Antolović, L. Stoimenov, "Visual Analysis in Reference Management Software", *Applied Mechanics and Materials*, vol. 197, pp. 633-637, 2012.
- [3] L. Stoimenov, N. Veljković, S. Bogdanović-Dinić, "Web application for scientific reference management", 10th International Conference on Telecommunication in Modern Satellite Cable and Broadcasting Services (TELSIKS), pp. 369-372, 2011.
- [4] Pdf-extract, <http://labs.crossref.org/pdfextract/>
- [5] D. Tkaczyk, P. Szostek, M. Fedoryszak, P.J. Dendek, L. Bolikowski, "CERMINE: automatic extraction of structured meta-data from scientific literature", *International Journal on Document Analysis and Recognition (IJ DAR)*, vol.18, no.4, pp. 317-335, 2015.
- [6] P. Lopez, "GROBID: Combining automatic bibliographic data recognition and term extraction for scholarship publications", *Research and Advanced Technology for Digital Libraries*, pp. 473-474, 2009.
- [7] J. Zou, L. Daniel, G.R. Thoma, "Locating and parsing bibliographic references in HTML medical articles", *International Journal on Document Analysis and Recognition (IJ DAR)*, vol. 13, no. 2, pp. 107-119, 2010.
- [8] G. Chowdhury, "Template mining for information extraction from digital documents", *Libr. Trends*, vol. 48, no.1, pp.182-208, 1999.
- [9] M. Day, T. Tsai, C. Sung, C. Lee, S. Wu, C. Ong, W. Hsu, "A knowledge-based approach to citation extraction", In *Information Reuse and Integration*, IEEE International Conference, pp. 50-55, 2005.
- [10] D. Tkaczyk, A. Collins, P. Sheridan, J. Beel, "Machine Learning vs. Rules and Out-of-the-Box vs. Retrained: An Evaluation of Open-Source Bibliographic Reference and Citation Parsers", In *Proceedings of the 18th ACM/IEEE on Joint Conference on Digital Libraries*, pp. 99-108, 2018.
- [11] B. Cui, X. Chen, "An improved hidden Markov model for literature metadata extraction", *Advanced Intelligent Computing Theories and Applications*, pp. 205-212, 2010.
- [12] K. Seymore, A. McCallum, R. Rosenfeld, "Learning hidden Markov model structure for information extraction", *AAAI-99 Workshop on Machine Learning for Information Extraction*, pp. 37-42, 1999.
- [13] I. Councill, C. Giles, M. Kan, "ParsCit: an open-source CRF reference string parsing package", In *Proceedings of the 6th International Language Resources and Evaluation*, 2008.
- [14] T. Okada, A. Takasu, J. Adachi, "Bibliographic component extraction using support vector machines and hidden Markov models", In *Proceedings of the ECDL*, pp. 501-512, 2004.
- [15] ORCID, <https://orcid.org/>
- [16] J. Kessler, M.K. Van Ullen, "Citation Generators: Generating Bibliographies for the Next Generation", *Journal of Academic Librarianship*, vol. 31, no. 4, pp. 310-316, 2005.
- [17] Refbase, <http://www.refbase.net>.
- [18] J.M.R. Gairin, P.R. Gallo, A.S. Duesa, "A Virtual Exhibition of Open Source Software for Libraries", 16th BOBCATSSS Symposium, Conference Proceedings, Zadar, pp. 319-325, 2008.
- [19] Aigaion, <http://www.aigaion.nl>.
- [20] S. Jose, F. Jayakanth, "Aigaion: A Web-based Open Source Software for Managing the Bibliographic References", *Journal of Information Science and Technology*, vol. 1, no. 1, pp.81-88, 2008.
- [21] RefWorks, <http://www.refworks.com>.
- [22] R. Gilmour, L. Cobus-Kuo, "Reference Management Software: A Comparative Analysis of Four Products", *Issues in Science and Technology Librarianship*, vol.66, pp. 63-75, 2011.
- [23] I.C. Hendrix, "RefWorks", *Journal of the Medical Library Association*, vol. 92, no. 1, pp. 111-113, 2004.
- [24] Wikindx, <http://wikindx.sourceforge.net>.
- [25] Memento, <http://findnwrite.com/memento>.
- [26] Crossref Metadata Search, <https://search.crossref.org/>
- [27] Scopus, <https://www.scopus.com>
- [28] The Rules of Procedure and the Manner of Evaluation and Quantitative Presentation of Scientific-research Results of Researchers, March 21st 2008, the Official Gazette of the Republic of Serbia, no. 18/10.
- [29] Citation Styles: APA, MLA, Chicago, Turabian, <https://pitt.libguides.com/citationhelp>
- [30] T. Ah-Hwee, "Text mining: The state of the art and the challenges". In: *Proceedings of the PAKDD 1999 Workshop on Knowledge Discovery from Advanced Databases* vol. 8, pp. 65-70, 1999.